

SYSTEM FOR CUSTOMIZING WEB PAGE

BACKGROUND OF THE INVENTION

Technical Field

5

10

15

20

The present invention relates to customization of a Web page, particularly to a method and a system for customizing a Web page which allows a user to add and erase an object on a Web page and to change an attribute of an object. Meanwhile, in this specification, addition/erasure of an object on a Web page and change of an attribute of an object are called customization. Moreover, an individual operation for customizing, addition of an object for instance, is called a customizing operation. A "user" is used as a word as opposed to an "author" who creates and manages an HTML document, and it means all the people who browse an HTML document (including end users and network administrators).

Prior Art

25

30

If a new hyperlink is to be added to an existing Web page, or if a link target of an existing hyperlink is to be changed, it is necessary to edit an HTML (Hyper Text Markup Language) file. It is also necessary to edit an HTML file likewise when an object such as text, an image, a table or a Java applet is added or erased on a Web page. However, since only a creator (an author) of an HTML file is usually authorized to edit the file, a user has not been allowed to add various information nor change a link target of a hyperlink. It is possible to save

HTML file on a hard disk by utilizing Web page capturing software and manually edit so as to modify a Web page. However, it requires high-level knowledge such as HTML and CS (Cascading Style Sheet), and in addition,
5 modification cannot be shared with other users since the modified page is saved on a local disk. Even if the modified HTML file is saved on a Web server, a new HTML file must be created each time it is modified so that the user needs to remember a URL for each modified HTML file
10 in addition to an original URL, and thus administration of URLs becomes complicated.

For instance, a community supporting system can be constructed with NetCollage of NEC. It is capable of
15 pasting text, an image, a program and an ink annotation directly on a Web page and also sharing the pasted objects with other users. Added objects are placed on a layer (a layer by an original technology, and not a layer of DynamicHTML), and it is possible to set what should be
20 displayed to which user by having access control on each layer. However, it is necessary to use a dedicated browser. Addable objects are only text, an image, a program and an ink annotation. It is not capable of changing an attribute of an object or erasing an existing
25 object. It is also necessary to install the program on a client machine in advance.

Therefore, an object of the present invention is to provide a method and a system for customizing a Web page
30 which does not require use of a dedicated browser.

Another object is to provide a method and a system for customizing a Web page which are capable of adding all types of objects describable in HTML such as text, an image, a program, a hyperlink, a table and a Java applet.

5

A further object is to provide a method and a system for customizing a Web page which are capable of changing an attribute of an object and erasing an existing object.

10

A still further object is to provide a method and a system for customizing a Web page which do not need to be installed on a client machine in advance.

15

A still further object is to provide a method and a system for customizing a Web page which allow addition, erasure and modification to be shared in a group.

20

A still further object is to provide a method and a system for customizing a Web page which allow a customized Web page to appear only by specifying an original URL.

SUMMARY OF THE INVENTION

25

To achieve the above-mentioned objects, by using at least one computer on which a browser for browsing a Web page runs, a Web page is customized by requesting a Web page to be customized, receiving a requested Web page in which a program for customizing a page is embedded, displaying the received Web page by a browser, having the program display a control panel for a customizing operation, and customizing a Web page according to a customizing

30

operation by a user using the control panel. For instance, if embedding of the program is done by a proxy server, it is organized so that a page is customized by requesting a page to be customized from a user's computer via a proxy server, having the proxy server embed a program for customization in the requested page, displaying the page in which the program is embedded by a browser of a computer, having the program display a control panel for a customizing operation, and customizing a page according to a customizing operation by a user using the control panel. And the data on the customizing operation is stored on the server. Moreover, a program for customization can be a program embedded in a locally (such as a hard disk, a diskette or a CD-ROM) stored Web page.

313
M/ Cooperative work for editing a page is performed by first requesting a page from another computer via a proxy server (hereafter, a "server"), having the server embed a program in the requested page, displaying the page in which the program is embedded by a browser of a computer, having the program send to a server information of the page, receiving data of a customizing operation already performed on the page, and having the program, based on this data, reflect a customizing operation performed on the page on a browser. And the program organizes it so that a control panel for a customizing operation is displayed, and from another computer, a page is further customized according to a customizing operation by a user using the control panel. Meanwhile, it is preferable to organize it so that the program to be embedded is not dependent on any operating system nor

application and the system immediately reflects a result of a customizing operation on a browser.

5 Data on customization is information of a customizing operation on a Web page and is managed separately from an original HTML file. Information of a customizing operation is retained together with a customizing user's ID, a URL of a customized Web page, and data on access rights to be managed for each customizing operation. A
10 customizing operation is dynamically reflected after displaying an original Web page. An object to be added to a Web page is also described as HTML elements. A program which runs on a client machine is described in Java and JavaScript.

15 To customize a Web page on a browser, functions of DynamicHTML (hereafter, a "DHTML") are used. When saving data of customizing operation, information of added/changed objects such as a URL of the Web page and a
20 position of an added object in a window is recorded and saved on a local disk or server. In the case of saving it on a server, it is possible to add information of whether or not to show it to other users so as to control to whom it is displayed. Meanwhile, access to a local
25 disk or a server is performed by a Java applet. To see a customized Web page, the Web page is loaded and then the dynamically customizing operations are reflected by a Java applet. To reflect customizing operations, functions of DHTML are used. Thus, it is also possible
30 to dynamically switch which customizing operation to be reflected on a Web page.

As it is organized in this way, it becomes possible for a user to freely add on an existing Web page text, a hyperlink, an image, a table and a Java applet and the like (hereafter, "objects") without changing an original HTML file. The added objects may also be moved or erased later. Without changing an original HTML file, a user may change an attribute of an object of an existing Web page. An attribute of an object referred to here is a link target of a hyperlink, a font or a color of text. Without changing an original HTML file, a user may erase an object on an existing Web page. Since information of a URL and changes made (for instance, types and positions of added objects) is saved, it is possible to display likewise later when the page is seen. Since addition and change of an object is managed separately from an original HTML file, other users are not affected. Namely, other users can see an unchanged original Web page. It is also possible, as for users in a group, to control what change is displayed depending on a user. As it is not editing an original HTML file but dynamically reflecting a customizing operation after displaying an original Web page, it becomes possible to dynamically switch whether to reflect it on a page for each customizing operation. Changes made can be controlled entirely or one by one as to display or non display. Accordingly, unlike a case of changing an original, views of both before and after a change may be provided. Since it is entirely described in Java and JavaScript, it is no longer necessary to install software other than a browser on a client machine.

It is also possible to share objects such as added text and images among members of a group so as to use a Web page as a bulletin board. It is also possible to control what change is displayed depending on a user. Since control of a hyperlink (addition, erasure, and change of a link target) can be implemented, it becomes possible for an administrator to control a link to an undesired page.

BRIEF DESCRIPTION OF THE DRAWINGS

Preferred embodiments of the present invention will now be described, by way of example only, with reference to the accompanying drawings in which:

Fig. 1 is a drawing showing an overview of a system for customizing a page of the present invention.

Fig. 2 is a drawing showing an example of a control panel for adding an object.

Fig. 3 is a drawing showing an example of an original Web page.

Fig. 4 is a drawing showing an example of adding text, a hyperlink, and an image to an original Web page.

Fig. 5 is a flowchart up to completion of customization of a page.

Fig. 6 is a flowchart in which, after customization is completed, users share it to perform editing work.

Fig. 7 shows an example of hardware configuration of a server and a computer for a user used in the present invention.

5
DETAILED DESCRIPTION OF THE
PREFERRED EMBODIMENTS OF THE INVENTION

Fig. 1 shows an overview of a system for customizing a page of the present invention.

10
<1> Downloading a Web page into a browser.

First, when a user downloads contents into a browser in (1) of Fig. 1, a customizer (a program for customization) is embedded in HTML by a proxy server in (2) to (3) of Fig. 1. Meanwhile, a customizer should preferably be a program written in Java and JavaScript.

15
<2> Reflecting customizing operations already performed on the Web page.

20
In (4) of Fig. 1, a customizer obtains a user ID and sends the user ID and a current URL to a database for customization. To obtain a user ID, there are methods such as using a cookie or using a control panel or the like to have a user directly enter an ID and a password (there may also be other methods, but they are not described in detail here since how to obtain a user ID is not the essence of the present invention). Moreover, in an environment where it is used not only by a group but by an individual, it is not necessary to obtain a user ID> In (5) of Fig. 1, out of the entire customizing operations performed to the URL, the customizer receives

from the database all of those whose access rights belong to the user. In (6) of Fig. 1, the customizer utilizes DHTML functions and customizes a Web page based on the received customizing operations. At this time, it is also possible to present a list of customizing operations to the user to select which operation to be reflected.

<3> A new customizing operation by the user on a browser.

In (7) of Fig. 1, the customizer displays the control panel and newly accepts a customizing operation from a user. In (6) of Fig. 1, the customizing operation designated by the user through the control panel is immediately reflected by the customizer on the browser. Moreover, the customizer stores information of customizing operations performed by the user in a database for customization in a timely manner. Data to be stored in the database includes an ID of the user who performed customization, a URL of the customized page, setting of access rights (to whom the operation is shown) and contents of the customizing operation and the like.

Fig. 5 shows a flowchart up to completion of the customization of the present invention. First, in step 510, the user requests a Web page to be customized of a Web server by using a browser. It can be the same method as the one used when browsing an ordinary Web page and requires no special operation. Next, in step 520, a proxy server embeds a program (a customizer) in an HTML file. A customizer is entirely written in Java and JavaScript. In step 540, the browser displays a page including the customizer. In step 540, the customizer

displays a control panel. The control panel in Fig. 2 is a user interface which accepts a designation of a customizing operation by the user. Next, in step 550, the user performs a customizing operation by using the control panel. For instance, pasting a new image on the page, or adding a new hyperlink. And on step 560, the customizer immediately reflects a customizing operation designated by the user on the browser. DHTML functions are used for reflecting it on the page. Lastly, in step 570, the customizer saves the customizing operation of the user. It can be saved either in a file on a local disk or in a database of the server. However, to share a customizing operation with members of a group, it needs to be saved in the database of the server.

Fig. 6 shows a flowchart in which, after customization is completed, users share it to perform editing work. First, in step 610, the user requests a web page to be customized of a Web server by using a Web browser. It can be the same method as the one used when browsing an ordinary Web page and requires no special operation. Next, in step 620, a proxy server embeds a customizer in an HTML file. In step 630, the browser displays a page including the customizer. At this point, the page before customization is displayed on the browser. In step 640, the customizer immediately sends information such as a URL to the server and received the data of a customizing operation already performed on this page. In step 650, the customizer reflects existing operations on the browser based on the received data. At this point, the page after customization is displayed on the browser. In step 660, the user can not only browse the customized

customized, how a customizer is embedded and passed to the browser. In customizing work, a user designates what change should be made by using a control panel. At this time, the control panel is provided by the customizer.

5 As Fig. 1 shows, a browser window can be used as a control panel, or a window can be made in Java. The change designated by the user is passed to the customizer, and the customizer reflects it on an HTML document on the browser by using DHTML functions. The
10 following describes methods for implementing each function of operations for customizing a Web page on a browser.

(i) Addition of an object

15 Fig. 2 shows an example of implementation of a control panel for addition of an object. It becomes possible for a user, by using the control panel in Fig. 2, to add text and a hyperlink on a Web page. Fig. 3 also shows an
20 original Web page. Fig. 4 shows an example of a Web page in which text, a hyperlink, and an image are added to the original Web page in Fig. 3. Fig. 4 shows the added hyperlink, added image, added text and a state in which text and a link are pasted like slips. Meanwhile, the
25 added image is an example of hiding the last link so as not to be clicked.

(ii) Change of an attribute of an object

Attributes of an object include a link target of a
30 hyperlink, a font, a size and a color of text. To change a link target, href property of a link object is modified.

547
(iii) Erasure of an object

To erase an object, a display property of an object can be changed so that it is no longer displayed.

The method for displaying a customized Web page on a browser is described further in detail. To display a customized Web page, after reading an HTML file into a browser, DHTML functions are utilized to reflect contents of customization. When downloading an HTML file, a customizer (JavaScript and Java applet) is embedded by a proxy server. The customizer finds customizing operations saved in a local file of a server or a client machine by retrieving them with a URL and a user ID as keys. It obtains the customizing operations and examines what change has been made to the Web page so as to reflect the customizing operations on a Web page by utilizing DHTML functions. When storing an additional operation of an object in a database for customization, if only coordinates on a browser window of the added object are recorded, in the case that each user has different setting of a font, a character size or the like, there are cases where a display position of the added object is different from what an original user intended. This is because a line feed position of text or a placement of a drawing changes depending on setting of a browser. To solve this problem, it should be recorded which object on a Web document the additional object was added onto, so it becomes possible to correctly display it even when browsers vary in setting.

Moreover, as to the Web services such as online shopping, there are sites where a technology for dynamically generating a page by a Web server and switching a

displayed advertisement for each customer is used. If such a personalized page is customized, and when another user sees that page, contents of the page may have changed and may not correctly reflect the customizing operation performed by the original user. To avoid such situation, there is a method for recording a last update data of a Web page simultaneously when storing information of a customizing operation in a database for customization. In this method, when a customizer reflects a customizing operation in (6) of Fig. 1, it can examine the last update date of the page and warn the user in case contents of the page at the time of the customizing operation may be different from those of the current page.

Fig. 7 shows an embodiment of hardware configuration of a server and a computer for a user used in the present invention. System 100 comprises a central processing unit (CPU) 1 and memory 4. CPU 1 and memory 4 are connected via bus 2, and by way of hard disk drive 13 (or a storage medium drive such as MO, CD-ROM 26 and DVD 32) and IDE controller 25 as auxiliary storage. Likewise, CPU 1 and memory 4 are connected via bus 2, and by way of hard disk drive 30 (or a storage medium drive such as MO 28, CD-ROM 29 and DVD 31) and SCSI controller 27 as auxiliary storage. Floppy disk drive 20 is connected with bus 2 via floppy disk controller 19. Preferably, a database for customization is stored in auxiliary storage.

A floppy disk is inserted into floppy disk drive 20, and a code or data of a computer program for giving

instructions to a CPU and so on to interact with an operating system to implement the present invention can be recorded on this floppy disk and the like, hard disk drive 13 (or a storage media such as MO, CD-ROM and DVD) and ROM 14, which is executed by being loaded to memory 4. The computer program comprises applications including an OS and a browser and other programs. The code of this computer program can also be compressed or divided into two or more so as to be recorded on two or more media.

System 100 can further have user interface hardware and comprise pointing device (a mouse, a joystick or the like) 7 or keyboard 6 for entry and display 12 for providing a user with visual data. It is also possible to connect with a printer via parallel port 16 or connect with a modem via serial port 15. This system 100 is connected with a network via serial port 15 and a modem or communication adapter 18 (Ethernet or Token-ring card) or the like so as to communicate with other computers and so on. For instance, it communicates with a Web server, proxy server and the like. Furthermore, it is possible to connect a remote transmitter-receiver with serial port 15 or parallel port 16 so as to exchange data by means of an infrared ray or an electric wave.

Speaker 23 receives a speech signal which is D/A (digital/analog) converted by audio controller 21 via amplifier 22 and outputs it as speech. In addition, audio controller 21 makes it possible to A/D (analog/digital) convert speech information received from microphone 24 and capture into the system speech information outside the system.

Thus, it may be easily understood that a server or a computer for a user in the present invention is implementable by a communication terminal with a communication facility including an ordinary personal computer (PC) or a workstation, a notebook PC, a palmtop PC, a network computer, various household electrical appliances with a built-in computer such as a TV set, a game console with a communication facility, a telephone, a fax, a portable telephone, a PHS, an electronic organizer or combination of these. However, these components are taken as examples, and it does not follow that all of them are required components of the present invention.

The present invention enables a user to freely add and erase an object on an existing Web page without changing an original HTML file. It enables a user to change an attribute of an object on an existing Web page without changing an original HTML file. Since addition, erasure and change of an object are managed separately from an original HTML file, other users are not affected. In addition, it is no longer necessary to install software other than a browser on a client machine.

While the invention has been particularly shown and described with respect to preferred embodiments thereof, it will be understood by those skilled in the art that the foregoing and other changes in form and details may be made therein without departing from the spirit and scope of the invention.